bio-protocol Preprint

Not Peer Reviewed
Ask a question online

Colorectal cancer microbiome meta-analysis

FB Francesco Beghini CH Curtis Huttenhower EF Eric A Franzosa NS Nicola Segata PM Paolo Manghi

Updated date: Oct 7, 2021

An abbreviated version of this protocol was published in eLIFE in May 2021

Integrating taxonomic, functional, and strain-level profiling of diverse microbial communities with bioBakery 3

DOI: 10.7554/eLife.65088

Detailed protocol

Dear Jessie,

MetaPhlAn 3 default output is made of relative abundances: therefore dividing each by 100 gives a representation of the probability of each bug. Keeping this in mind, you can easily compute Shannon entropy (as Shannon diversity) based on the standard formula: -sum(each probability * log(each probability)), excluding the zero values. If you don't feel confident you can easily use functions in scipy or scikit-bio, or test yours against these.

For what concerns rarefaction, the most straight forward method is to directly subset the fastq file of each sample down to the desired number of reads: you can count the number of reads in the file with a fast function:

```
def rawpycount(filename):
    f = openr(filename,'rb')
    f_gen = _make_gen(f.read)
    return sum( buf.count(b'\n') for buf in f_gen)
```

and then printing out the 4 lines fastq randomly choosen:

```
def random_sample(par):
    N,n = rawpycount(par['inp_f'])/4, par['nreads']
    if N<n: exit(1)
    sample = (np.fromiter(np.random.choice(N,n,replace=False), dtype=np.int64))
    length_sample = len(sample)
    sample = set(sample)
    length, line = 0, 1
    f, i, c = openr(par['inp_f'],'r'), -1, 0

    while (line and length<length_sample):
        line = f.readline()
        if c%4==0:
            i += 1  ## read..
            if i in sample:
                length += 1
                print line.rstrip()
                for ii in range(3):
                    linetokeep = f.readline()
                    print linetokeep.rstrip()
                c += 3
        c+=1
```

Another method is to identify a percentage of reads that corresponds to the desired number, then use the "fake coin". Suppose that you want to rarefy to the 40%, for each fastq you generate a random number between 0 and 1. If it is <= 0.4, you preserve the read, otherwise you discard it. This method can be applied also to the bowtie output from metaphlan, making you gaining time.

```
matching_reads_subset = [ m for m,rand in zip( bowtie_matching_reads, np.random.rand( tot_matches )) if rand<=choosen_percentage_wrt_min ]
```

Bye

How to cite:(Readers should cite both the Bio-protocol preprint and the original research article where this protocol was used)

1. Beghini, F. , Huttenhower, C. , Franzosa, E. , Segata, N. and Manghi, P. (2021). Colorectal cancer microbiome meta-analysis. Bio-protocol Preprint. bio-protocol.org/prep1397.
2. Beghini, F., McIver, L. J., Blanco-Míguez, A., Dubois, L., Asnicar, F., Maharjan, S., Mailyan, A., Manghi, P., Scholz, M., Thomas, A. M., Valles-Colomer, M., Weingart, G., Zhang, Y., Zolfo, M., Huttenhower, C., Franzosa, E. A. and Segata, N.(2021). Integrating taxonomic, functional, and strain-level profiling of diverse microbial communities with bioBakery 3. eLIFE. DOI: 10.7554/eLife.65088

Copyright: Content may be subjected to copyright.